ITERATIVE GENERALISED WIENER FILTERING OF NOISY AND DEGRADED IMAGES

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

by
MRITYUNJOY CHAKRABORTY

to the
DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
JUNE, 1985

DEDICATED TO

MY PARENTS

CERTIFICATE

Certified that the work presented in this thesis entitled 'ITERATIVE GENERALISED WIENER FILTERING OF NOISY AND DEGRADED IMAGES' by M. Chakraborty has been carried out under my supervision and that this has not been submitted elsewhere for a degree.

SKMullick

June, 1985.

DR. S. K. MULLICK
Professor
Department of Electrical Engineering
Indian Institute of Technology
Kanpur-208016
India

ACKNOWLEDGEMENT

I take this opportunity to express my indebtedness and sincerest thanks to my thesis supervisor, Prof. S.K. Mullick for providing me invaluable and unfailing guidance throughout the course of this work.

I express my profound sense of gratitude to all my instructors who exposed me to so many fascinating areas of digital signal and image processing. Also, I gratefully acknowledge the useful discussions I had with my unforgettable friend, Mr. P.G. Poonacha, the research scholar of Electrical Engineering.

I record my heartiest thanks to all my friends who made my stay at IIT-Kanpur a memorable one and also to those whose inspirations I can never forget in my future course of life.

Finally, I thank Mr. Yogendra for his sincere typing work.

Mrityunjoy Chakraborty

ABSTRACT

An iterative procedure of Wiener filtering for the restoration of images proposed in (2,7) is studied in detail for the restoration of images degraded by imaging systems and further corrupted by additive and multiplicative noise. It is further extended to include generalised Wiener filtering and suboptimal filtering in other than frequency domain in order to reduce the number of computations. Simulation results support the effectiveness of the iteration strategy and shows that the mean square error converges in few iterations (typically three to four) and good restorations are obtained. The iterative procedure overcomes some of the shortcomings associated with Wiener filtering.

CONTENTS

CHAPTER 1

IMAGE RESTORATION

Digital restoration of pictures deals with images which are degraded by imaging systems and further contaminated by extraneous noise. In some cases, the degradation is a 'point degradation' process where only the gray levels of the individual picture points are affected without introducing blur. Other types which involve blur are known as 'spatial degradations'. Still further degradation arises from temporal or chromatic effect. Here we will consider only the first two processes which corrupt a picture. These occur in various applications. For example, in astronomy and remote sensing, the pictures obtained are degraded by atmospheric turbulence, aberrations of the optical system and relative motion between the object and the camera. For medical radiographic images, both the resolution and contrast are poor due to the nature of the X-ray imaging systems. Electron micrographs are often degraded by the spherical aberration of the electron lens. One also encounters non-linear imaging systems. However restoration in non-linear models requires enormous computational effort and has not achieved any significant progress. An a result the assumption of linearity is usually made. Linear imaging systems may be shift variant/invariant. Further the image can be contaminated by both additive and multiplicative, signal dependent and independent noise during detection and recording p rocess.

In the following, we will consider the presence of both additive and multiplicative noise.

IMAGE FORMATION AND RECORDING MODEL :

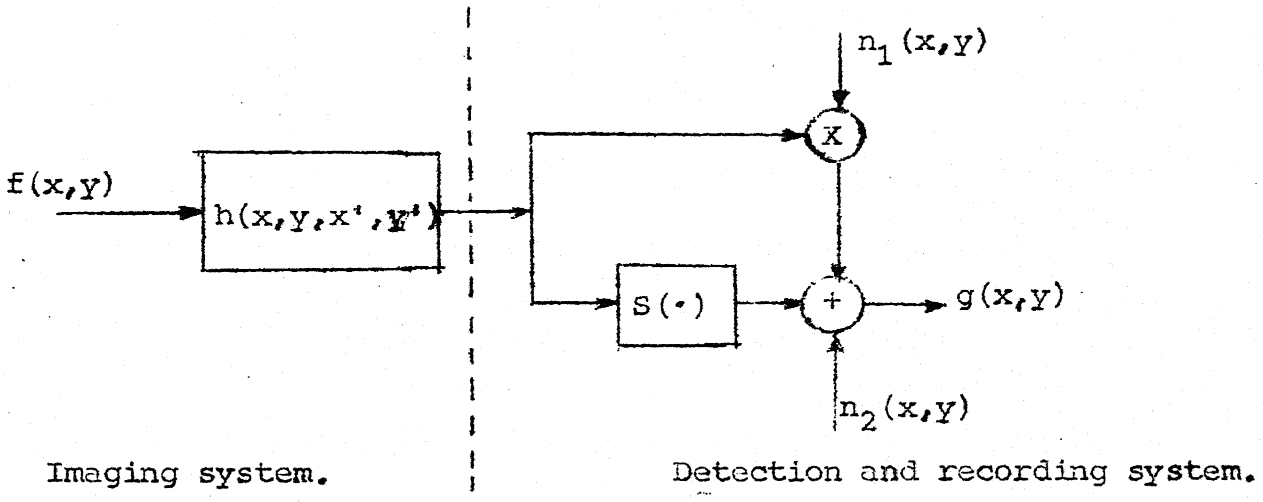


Fig. 1.1 Continuous image formation and recording model

Fig. 1.1 shows a comprehensive model (2) of the imaging and recording system for continuous signals (The discrete version of this model is quite straightforward and will be taken up in Chapter 3). The following notations are used in this discussion.

$f(x,y)$ = Gray level function of the original object.

$h(x,y,x',y')$ = Point spread function/Impulse response of the imaging system i.e. the output of the imaging system at (x,y) corresponding to an impulse input described by $f(x,y) = \delta(x-\alpha, y-\beta)$ will be given by $h(x,y,\alpha,\beta)$. The PSF (Point-spread function) is, in general, dependent on the position (α,β) of the impulse in the original picture.

$S(\cdot)$ = Detector response function. In the following discussion, we have assumed it to be identity transformation.

$n_1(x,y)$ = Multiplicative noise-component.

$n_2(x,y)$ = Additive noise-component.

The final output $g(x,y)$ is related to the input object by

$$g(x,y) = \iint h(x,y,x',y')f(x',y')\,dx'\,dy' + n_1(x,y) \iint h(x,y,x',y')f(x',y')dx'dy' + n_2(x,y). \quad .. (1.1)$$

If, however, the degrading system is space-invariant, then the PSF takes the form $h(x-x', y-y')$ and accordingly,

$$g(x,y) = \int\int h(x-x',y-y')f(x',y')dx'dy'+n_1(x,y)\int\int h(x-x',y-y') f(x',y')dx'dy' + n_2(x,y) \qquad .. (1.2)$$

The objective of the estimation procedure is to make as good an estimate of the original picture as possible given the model (Fig. 1.1) and the degraded picture. Any such estimation procedure requires some knowledge concerning both the degradation function and the noise. In some cases, the physical phenomenon governing the degradation can be used to determine the PSF. In other situations, the PSF may be found out from the degraded output itself with a test-picture at the input (4). Apart from the PSF, also required are the statistical properties of the noise and the knowledge about how it is correlated with the picture. The most common assumption about noise is that it is white i.e. its spectral density is constant over all frequencies and it is uncorrelated with the picture. The concept of white noise is a mathematical abstraction and though not always admissible, it is a reasonable assumption provided the noise bandwidth is much larger than the picture bandwidth.

In what follows, we will start with the restoration of one-dimensional signals. Modifications required for two-dimensional applications are straightforward and will be explained later on.

CHAPTER 2

CONTINUOUS ITERATIVE WIENER FILTER

The restoration of images in a continuous model will be considered in this chapter. We will assume the degrading system to be deterministic and space invariant. Also we will restrict ourselves to the restoration of one-dimensional signals in this chapter.

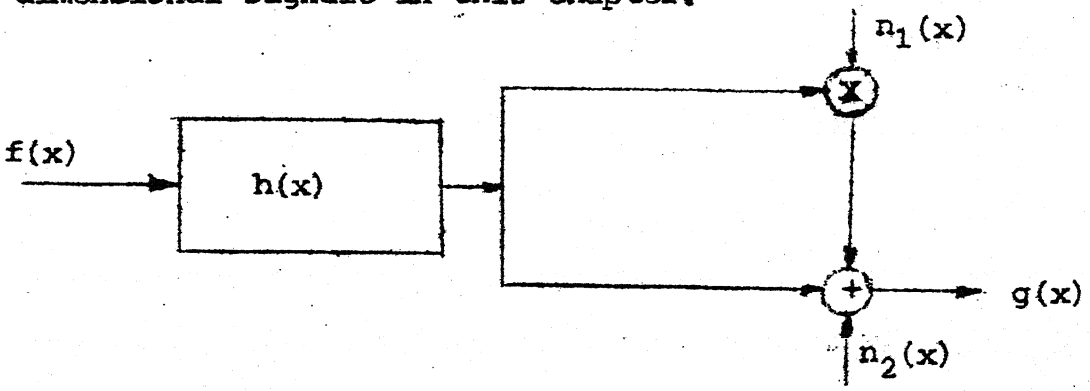


Fig. 2.1 Imaging and recording model for one-dimensional signals with space-invariant degrading system.

Fig. 2.1 shows the corresponding imaging model with h(x) denoting the impulse response of the space-invariant degrading system. The output g(x) is given by

$$g(x) = \int h(x-x')f(x')dx' + n_1(x) \int h(x-x')f(x')dx' + n_2(x) \quad .. (2.1)$$

Fourier-transforming both sides,

$$G(jw) = H(jw).F(jw) + N_1(jw)* (H(jw).F(jw)) + N_2(jw) \quad .. (2.2)$$

where $G(jw)$, $H(jw)$, $F(jw)$, $N_1(jw)$ and $N_2(jw)$ are the Fourier-transforms of $g(x)$, $h(x)$, $f(x)$, $n_1(x)$ and $n_2(x)$ respectively and '*' denotes convolution.

In the absence of noise, we have

$$G(jw) = H(jw).F(jw)$$

or equivalently,

$$F(jw) = G(jw)/H(jw) \qquad .. (2.3)$$

which implies that f(x) can be restored by multiplying the Fourier transform of the degraded output by the function $\frac{1}{H(jw)}$ and then inverse Fourier transforming i.e. the filter function is $\frac{1}{H(jw)}$. This is known as 'Inverse filter'. However, a number of problems arise when one attempts to make practical use of it (4). There may be points on the w-axis where H(jw) is zero and consequently G(jw) is also zero, resulting in indeterminate ratios. In the presence of noise, the zeros of G(jw) do not coincide with those of H(jw). As a result, the division by H(jw) would lead to very large values in the vicinity of the zeros of H(jw). In fact, when noise is present, we have

$$G(jw) = H(jw).F(jw) + N_1(jw)*(H(jw).F(jw)) + N_2(jw)$$

$$\text{or,}\quad \frac{G(jw)}{H(jw)} = F(jw) + \frac{(N_1(jw)*(H(jw).F(jw)))}{H(jw)} + \frac{N_2(jw)}{H(jw)} \quad .. (2.4)$$

When H(jw) is very small, the two terms

$$\frac{(N_1(jw)*(H(jw).F(jw)))}{H(jw)}$$

and
$$\frac{N_2(jw)}{H(jw)}$$

may become much larger in magnitude than F(jw). The inverse transform then no longer resembles f(x), but contains many noise-like variations.

The difficulty seems to be that the form of the filter is fixed in the case of inverse filter. If, however, the filter is obtained by an estimation procedure where some measure of the difference between the restored output and the original input is minimized in a statistical framework, some of the drawbacks of the inverse filter can be avoided. The most common measure is the mean square error between the input and the output. The corresponding filter is known as <u>Wiener</u>[1] <u>filter</u>.

Here we will assume that the original input and both the additive and multiplicative noise components i.e. $f(x)$, $n_2(x)$ and $n_1(x)$ respectively belong to zero-mean, mutually independent random processes.

As described by equation (2.1), the output $g(x)$ is given by

$$g(x) = \int h(x-x')f(x')dx' + n_1(x) . \int h(x-x')f(x')dx' + n_2(x)$$

This equation can not be solved directly when noise is present, since both $n_1(x)$ and $n_2(x)$ are not known to us though their statistical properties are assumed to be known. Instead, one can try to minimize the mean. sq. error

$$e^2 = E \{ (f(x) - \hat{f}(x))^2 \} \quad .. (2.5)$$

in order to get the least-sq-estimate $\hat{f}(x)$ of $f(x)$, given $g(x)$ (E denotes the expectation value). The unconstrained solution of this minimization is evidently the conditional expectation of $f(x)$ given $g(x)$ (9), which is, in general, a non-linear function of $g(x)$. Also one requires the joint probability density over the random processes $f(x)$ and $g(x)$ making the process more complicated and inconvenient.

The problem becomes simplified if it is assumed that the estimate $\hat{f}(x)$ is a linear function of the gray levels in $g(x)$.

The corresponding estimate is known as the linear least square estimate.

Mathematically, this is expressed by

$$\hat{f}(x) = \int m(x,x') \; g(x') \; dx' \qquad .. \; (2.6)$$

where $m(x,x')$ is the weight to be applied to $g(x)$ at $x = x'$ for the computation of $\hat{f}$ at x. We assume that both $f(x)$, $n_1(x)$ and $n_2(x)$ are second order stationary processes. In such a case, the weighting function depends only om $(x-x')$ and we have

$$\hat{f}(x) = \int m(x-x') \; g(x') \; dx' \qquad .. \; (2.7)$$

and so

$$e^2 = E \; \{ (f(x) - \int m(x-x')g(x')dx')^2 \} \qquad .. \; (2.8)$$

e^2 can be minimized by several standard procedures (e.g. variational calculus techniques) and one obtains the following integral equation (4).

$$\int m(x-x') \; R_{gg}(x'-\gamma)dx' = R_{fg}(x-\gamma) \qquad .. \; (2.8)$$

where $R_{gg}(x-\tau) = E(g(x)g(\tau))$ = Autocorrelation of $g(x)$.

and $R_{fg}(x-\tau) = E(f(x).g(\tau))$ = Cross-correlation of $f(x)$ and $g(x)$.

From equation (2.8), one finally obtains (2) the filter function (in frequency domain) $M(jw)$ as

$$M(jw) = \frac{1}{H(jw)} \left[\frac{|H(jw)|^{2} \, \Phi_f(w)}{|H(jw)|^{2} \, \Phi_f(w) + (|H(jw)|^{2} . \Phi_f(w)) * \Phi_{n_1}(w) + \Phi_{n_2}(w)} \right] \quad .. \ (2.9)$$

('*' denotes convolution.)

where $\Phi_f(w)$ = Power-spectral density of the input object $f(x)$,

$\Phi_{n_1}(w)$ = Power-spectral density of the multiplicative noise $n_1(x)$.

and $\Phi_{n_2}(w)$ = Power-spectral density of the additive noise $n_2(x)$.

(The PSD (Power spectral density), by definition, is the Fourier transform of the corresponding autocorrelation function).

The $M(jw)$ as given by equation (2.8) is the well-known Wiener filter for the model under discussion.

It is observed that the Wiener filter is indeed the inverse filter multiplied by a noise smoothing factor M'(w) where

$$M'(w) = \frac{|H(jw)|^2 \Phi_f(w)}{|H(jw)|^2 \Phi_f(w)+(|H(jw)|^2 \Phi_f(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)} \quad .. (2.10)$$

In noise-less environment i.e. when $\Phi_{n_1}(w) = \Phi_{n_2}(w) = 0$, the Wiener filter is exactly the inverse filter. A close examination of equation (2.9) shows that the denominator actually represents the PSD of the recorded image and thus can be computed by any of the standard power-spectrum estimation techniques. However, the evaluation of the numerator requires the explicit knowledge of $\Phi_f(w)$ i.e. object PSD and thus one requires an object prototype for the computation of $\Phi_f(w)$. Mathematical convenience is the main reason for the choice of the minimization of m.s.e. (mean square error) as the criterion for filter design and unless a clear and reasonably accurate model of human visual process is established, this is likely to be a popular criterion.

However, as already explained, the filter requires an extensive knowledge of the PSD of the object, or its autocorrelation function. Though the PSD of the noise can be obtained from a suitable model, that of the object is essentially an intrinsic property of the object itself. In addition, Wiener filter produces the optimum estimate only on an average. Since it solves an unconstrained minimization problem, false details due to negative pixel values are also encoantered (2,7).

ITERATIVE WIENER FILTER (CONTINUOUS) :

An iterative algorithm has been proposed and studied by Ramakrishna R.S. (2,7) which dispenses with the extensive knowledge of the object PSD for the computation of the Wiener filter (Fig. 2.2).

With an initial guess $\Phi_{1f}(w)$ of the object PSD, the Wiener filter is obtained as

$$M(jw) = \frac{1}{H(jw)} \left[\frac{|H(jw)|^2 \quad \Phi_{1f}(w)}{|H(jw)|^2 \; \Phi_{1f}(w) + (\,|H(jw)|^2 \; \Phi_{1f}(w))*\Phi_{n_1}(w) + \Phi_{n_2}(w)} \right] \qquad .. (2.11)$$

$$= \frac{1}{\dfrac{|H(jw)|^2 . \Phi_{1f}(w)}{\Phi_{1f}(w) . H^*(jw)} + \dfrac{(\,|H(jw)|^2 \Phi_{1f}(w))*\Phi_{n_1}(w)}{\Phi_{1f}(w) . H^*(jw)} + \dfrac{\Phi_{n_2}(w)}{\Phi_{1f}(w) . H^*(jw)}}$$

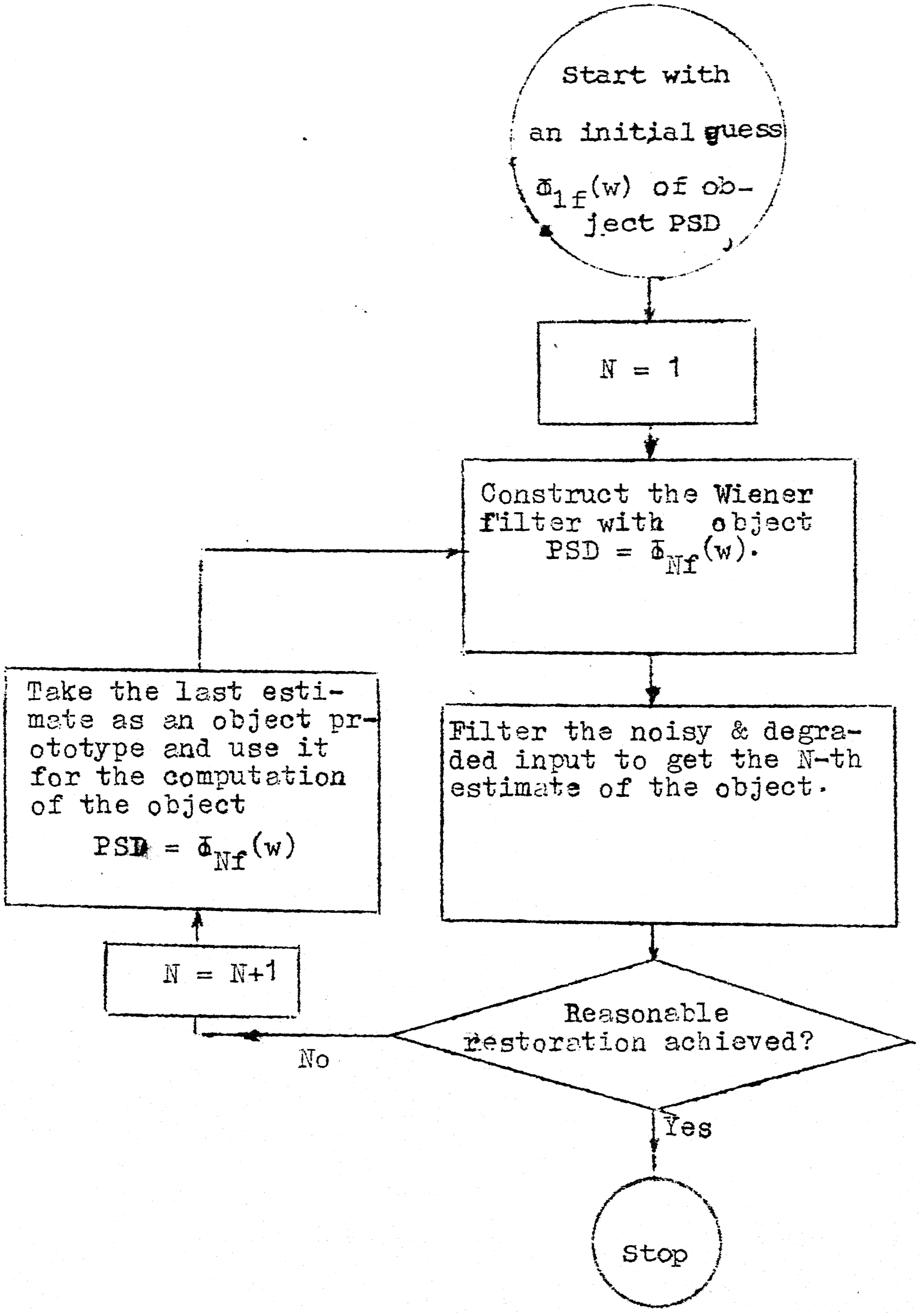

Start with an initial guess $\Phi_{1f}(w)$ of object PSD
N = 1
Construct the Wiener filter with object PSD = $\Phi_{Nf}(w)$.
Filter the noisy & degraded input to get the N-th estimate of the object.
Reasonable restoration achieved?
No
N = N+1
Take the last estimate as an object prototype and use it for the computation of the object PSD = $\Phi_{Nf}(w)$
Yes
Stop

Fig. 2.2 Iterative algorithm

If $\Phi_o(w)$ and $\Phi_i(w)$ be the PSD-s corresponding to the restored and recorded images, then they are related by

$$\Phi_o(w) = |M(jw)|^2 \, \Phi_i(w) \qquad .. \; (2.12)$$

where
$$\Phi_i(w) = |H(jw)|^2 \, \Phi_f(w) + (|H(jw)|^2 \, \Phi_f(w))*\Phi_{n_1}(w) + \Phi_{n_2}(w) \qquad .. \; (2.13)$$

As is shown below, with an appropriate guess of the object PSD for the computation of the Wiener filter, the PSD of the restored output closely resembles that of the original object.

From (2.11), (2.12) and (2.13), we get

$$\Phi_o(w) = \frac{|H(jw)|^2 \, \Phi_f(w)+(|H(jw)|^2 \, \Phi_f(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)}{\left[\left| \dfrac{|H(jw)|^2 \Phi_{1f}(w)}{\Phi_{1f}(w)H^*(jw)} + \dfrac{(|H(jw)|^2 \Phi_{1f}(w))*\Phi_{n_1}(w)}{\Phi_{1f}(w).H^*(jw)} + \dfrac{\Phi_{n_2}(w)}{\Phi_{1f}(w).H^*(jw)} \right|^2\right]}$$

$$= \frac{1 + \dfrac{(|H(jw)|^2 \Phi_f(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)}{|H(jw)|^2 \Phi f(w)}}{\left[1 + \dfrac{(|H(jw)|^2 \Phi_{1f}(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)}{|H(jw)|^2 \quad \Phi_{1f}(w)} \right]^2} \cdot \Phi_f(w)$$

$$= O(w).\Phi_f(w) \qquad .. (2.14)$$

where,

$$O(w) = \frac{1 + \dfrac{(|H(jw)|^2\Phi_f(w))*\Phi_{n_1}(w) + \Phi_{n_2}(w)}{|H(jw)|^2\,\Phi_f(w)}}{\left[1 + \dfrac{(|H(jw)|^2\,\Phi_{1f}(w))*\Phi_{n_1}(w) + \Phi_{n_2}(w)}{|H(jw)|^2\,\Phi_{1f}(w)}\right]^2} \qquad .. (2.15)$$

If both $|H(jw)|^2 \Phi_f(w)$ i.e. PSD of the degraded object and $|H(jw)|^2 \Phi_{1f}(w)$ are reasonably large compared to(($|H(jw)|^2 \Phi_f(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)$) and (($|H(jw)|^2 .\Phi_{1f}(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)$) respectively at each w (at least over most of the frequencies under consideration), then $O(w) \simeq 1$ and so $\Phi_o(w) \simeq \Phi_f(w)$,

(Note that (($|H(jw)|^2 \Phi_f(w))*\Phi_{n_1}(w)+\Phi_{n_2}(w)$) actually represents the PSD of the component of the degraded output contaminated by noise) which shows that the PSD calculated from the first iterate of this procedure is very close to the object PSD provided the initial guess is reasonable. If the input SNR is known to be quite high, one can compute the PSD of the recorded object and use it as the first guess. Also one can start with an appropriate model of the input PSD and calculate the parameters associated from the recorded object itself (e.g. in many situations, it is quite reasonable to use a lowpass filter function as the object power-spectral density, or an exponential autocorrelation function). In further steps of iteration, the PSD is calculated from the iterates themselves. Proceeding in this manner, one can reach a stage when significant improvement is reached and in an interactive situation, one can stop the process of iteration on obtaining reasonable restoration. This is well supported by computer results in the case of one-dimensional continuous signals in (2) and for discrete generalised Wiener filtering of two-dimensional data as described in the next chapter.

CHAPTER 3

DISCRETE ITERATIVE GENERALISED WIENER FILTER

In this chapter, we will consider the restoration of discrete signals. The corresponding model of the imaging and recording system is explained below (Fig. 3.1).

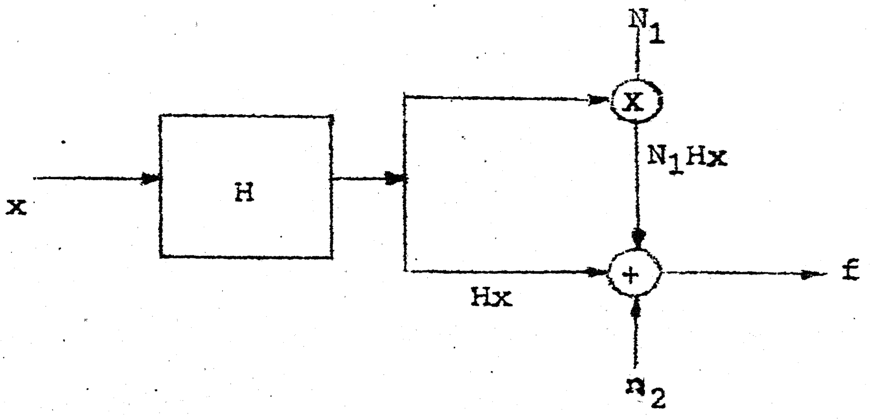


Fig. 3.1 Discrete image formation and recording system.

The following notations are used in this model.

x = A M-element data vector representing the original discrete data sequence.

H = A M X M transformation matrix corresponding to the degradation process of the imaging system. For a space-invariant system, H is a circulant matrix.

N_1 = A M X M diagonal matrix, where the diagonal elements belong to the multiplicative noise process.

n_2 = A M-element additive noise vector.

f = The M-element, degraded and noisy output vector.

(Note that in our notation, lower-case letters are used to represent vectors and upper-case ones for matrices.)

The output f is given by

$$f = Hx + N_1 \ Hx + n_2 \qquad .. \ (3.1)$$

As in the case of continuous Wiener filter, here also we assume that the input x and both the additive and multiplicative noise components belong to zero-mean, mutually independent random processes.

The discrete Wiener estimation of the input object essentially consists of finding a M X M filter matrix A which minimizes the mean square error between the original data sequence x and the restored sequence $\hat{x}$. The mean square error e^2 between x and $\hat{x}$ is given by

$$e^2 = E((x-\hat{x})^t (x-\hat{x})) \qquad .. \ (3.2)$$

A direct minimization of e^2 as given by equation (3.2) leads to a filter matrix A which can be used to filter the recorded data in space-domain. However, it is possible to perform the Wiener filter operation in different transform domains utilising unitary transforms and it is thereby possible to significantly reduce the computational complexities by selective computation. This is known as 'Generalised Wiener filtering' (1).

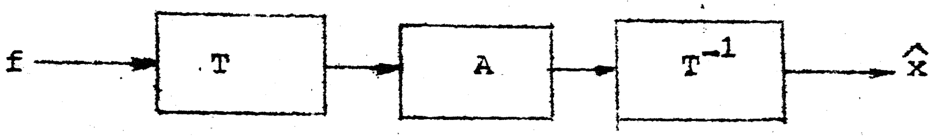


Fig. 3.2 Generalised Wiener filtering.

Fig. 3.2 shows the corresponding block-diagram of the generalised Wiener filter. First, a M X M unitary transform T is applied to the recorded data vector f, followed by the multiplication by the M X M filter matrix (in the transform domain) A and finally, the restored output $\hat{x}$ is obtained by taking the inverse transform of the filtered data. The restored output $\hat{x}$ is given by

$$\hat{x} = T^{-1} A T f \quad .. (3.3)$$

where $$T^{-1} = T^{*t} \quad .. (3.4)$$

since T is a unitary transform.

Let $$B = T^{-1} A T \quad .. (3.5)$$

Note that B is related to A by a similarity transformation (equation (3.5)) and so there is a one-to-one correspondence between A and B. As a result, one can minimize e^2 (equation (3.2)) w.r.t. B in order to compute the Wiener filter matrix A. Therefore

$$\hat{x} = Bf \qquad .. (3.6)$$

From equation (3.2) and (3.6),

$$\begin{aligned} e^2 &= E((x-\hat{x})^t(x-\hat{x})) \\ &= E(x^t x) - 2.\ E(x^t\hat{x}) + E(\hat{x}^t\hat{x}) \\ &= E(x^t x) - 2E(x^t Bf) + E(f^t B^t Bf) \qquad .. (3.7) \end{aligned}$$

Now B must be chosen in such a way that e^2 is minimum. We differentiate e^2 by the elements of the matrix B and equate them to zero i.e. we make $\nabla_B(e^2) = [0]$ where $[0]$ is the null matrix and ∇_B is the gradient operator, i.e.

$$\nabla_B(e^2) = \begin{bmatrix} \frac{\partial(e^2)}{\partial B_{11}} & \frac{\partial(e^2)}{\partial B_{12}} & \cdots\cdots & \frac{\partial(e^2)}{\partial B_{1M}} \\ \vdots & \vdots & & \vdots \\ \frac{\partial(e^2)}{\partial B_{M1}} & \frac{\partial(e^2)}{\partial B_{M2}} & \cdots\cdots & \frac{\partial(e^2)}{\partial B_{MM}} \end{bmatrix}$$

It is to be noted that ∇_B is a linear operator and so can be applied before taking the expectation value. We have the following identities :

$$\nabla_B(x^t B\ f) = xf^t \quad .. (3.8)$$

$$\nabla_B(f^t\ B^t\ Bf) = 2.B\ ff^t \quad .. (3.9)$$

and $$\nabla_B(x^t x) = 0 \quad .. (3.10)$$

From equations (3.7), (3.8), (3.9) and (3.10),

$$\nabla_B(e^2) = 0$$

or, $$E\left\{\nabla_B(x^t x)\right\} - 2E\left\{\nabla_B(x^t Bf)\right\} + E\left\{\nabla_B(f^t B^t Bf)\right\} = 0$$

which leads to

$$E.(Bff^t) = E(xf^t)$$

or, $$B.E(ff^t) = E(xf^t) \quad .. (3.11)$$

Now from equation (3.1),

$$f = Hx + N_1\ Hx + n_2$$

Hence $$E(xf^t) = E.(xx^t).H^t + E.(xx^t H^t N_1) + E.(xn_2{}^t)$$

Now, $E(xx^t) = R_x$ i.e. autocorrelation matrix of the random process x.

$E(xn_2^t) = [0]$, since x and n_2 are mutually independent zero-mean random processes.

and $E.(xx^tH^tN_1) = [0]$, since x and n_1 are also mutually independent, zero-mean random processes.

(This is evident if one considers any element, say (i,j) th element of $E.(xx^tH^tN_1)$, since $E(xx^tH^tN_1)_{ij} = E(\sum_K x_i x_K N_{1jj} H_{jK}) = 0)$

Thus $E(xf^t) = R_x H^t$ .. (3.12)

Also, $E(ff^t) = E((Hx+N_1Hx+n_2)(Hx+N_1Hx+n_2)^t)$

$$= E(Hxx^tH^t) + E\ (Hxx^tH^tN_1) + E\ (Hx\ n_2^t)$$

$$+ E(N_1Hxx^tH^t) + E\ (N_1Hxx^tH^tN_1) + E(N_1Hx\ n_2^t)$$

$$+ E(n_2x^tH^t) + E.(n_2x^tH^tN_1) + E(n_2n_2^t)$$

As before $E(xn_2^t) = E(n_2x^t) = 0$;

$E(n_2n_2^t) = R_{N_2}$ i.e. the autocorrelation matrix of n_2.

Also, since x, N_1 and n_2 are mutually independent, zero-mean random processes, the cross-terms $E(Hxx^tH^tN_1)$, $E(N_1Hxx^tH^t)$, $E(N_1\ Hx\ n_2^t)$ and $E\ (n_2x^tH^tN_1)$ are zero (For details, see the Appendix).

Now, $E(N_1 Hxx^t H^t N_1)_{ij}$

$$= E.(\sum_l \sum_k N_{1_{ii}} H_{il} x_l x_k N_{1_{jj}} H_{jk})$$

$$= E.(N_{1_{ii}} N_{1_{jj}} \sum_l \sum_k H_{il} x_l x_k H_{jk})$$

$$= E(N_{1_{ii}} N_{1_{jj}}). E(\sum_l \sum_k H_{il} x_l x_k H_{jk})$$

(since x and N_1 are mutually independent random processes)

$$= ((H R_x H^t) \odot R_{N_1})_{ij}$$

where R_{N_1} in the autocorrelation matrix of the multiplicative noise process and $\odot$ denotes the Hadamard product of matrices , defined by $(A \odot B)_{kj} = a_{kj} b_{kj}$, A and B being two matrices and a_{kj} and b_{kj} their (k,j)th elements respectively.

Thus $E(ff^t) = H R_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2}$ .. (3.13)

From equation (3.5), (3.11), (3.12) and (3.13),

$$B.E(ff^t) = E\ (xf^t)$$

or, $B.(H R_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2}) = R_x H^t$

or, $B = R_x H^t (H R_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2})^{-1}$

which leads to

$$A = T(R_xH^t(HR_xH^t+(HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1})T^{-1} \quad .. \; (3.14)$$

The matrix A, thus obtained from equation (3.14) is the <u>generalised Wiener filter matrix</u>. Note that the restored output $\hat{x}$ is given by

$$\begin{aligned}\hat{x} &= T^{-1} \, A \, T \, f \\ &= T^{-1}T \, (R_xH^t(HR_xH^t+(HR_xH^t) \odot R_{N_1}+R_{N_2})^{-1})T^{-1}Tf \\ &= (R_xH^t(HR_xH^t+(HR_xH^t) \odot R_{N_1}+R_{N_2})^{-1})f \quad .. \; (3.15)\end{aligned}$$

which is independent of T and so the mean square error $e^2 = E.((x-\hat{x})^t(x-\hat{x}))$ does not depend on the transform chosen. However, the nature of the filter matrix A is strongly dependent on T and thus one can choose a suitable transfo rm in order to minimize the computational complexities in filter generation and filter operation (1).

MINIMUM MEAN SQUARE ERROR :

The mean square error e^2 is given by (equation 3.2)

$$e^2 = E. \; ((x-\hat{x})^t(x-\hat{x})).$$

or equivalently,

$$e^2 = \text{Trace}\left[E\left\{ (x-\hat{x})(x-\hat{x})^t \right\} \right]$$

$$= \text{Trace}\ (E(xx^t)) - \text{Trace}\ (E(x\hat{x}^t)) - \text{Trace}\ (E(\hat{x}x^t))$$
$$+ \text{Trace}\ (E(\hat{x}\hat{x}^t))$$

$$= \text{Trace}\ (R_x) - 2.\text{Trace}\ (E(\hat{x}x^t)) + \text{Trace}\ (E(\hat{x}\hat{x}^t))$$

(Since $(x\hat{x}^t)$ is the transpose of $(\hat{x}x^t)$ and so trace remains invariant.)

$$= \text{Trace}\ (R_x) - 2.\text{Trace}\ (E(Bfx^t)) + \text{Trace}\ (E(Bff^tB^t))$$

(where B is given by equation (3.5))

$$= \text{Trace}\ (R_x) - 2.\text{Trace}\ (R_xH^t(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1}\ HR_x)$$
$$+ \text{Trace}\ (R_xH^t(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1}$$
$$(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2}).$$
$$(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1}\ HR_x)$$

(Using equations (3.12) and (3.13). Note that $(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1}$ is a symmetric matrix)

$$= \text{Trace}\ (R_x - R_xH^t(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1}\ HR_x)$$

.. (3.16)

The minimum mean square error is given by equation (3.16) and as stated earlier, it is seen that the minimum m.s.e is independent of the transform chosen. In the absence of degradation and multiplicative noise i.e. when $H = I$ (identity matrix) and $R_{N_1} = [0]$, the minimum m.s.e. is given by

$$e^2 = \text{Trace } (R_x - R_x(R_x + R_N)^{-1} R_x)$$

$$= \text{Trace } \left[R_x(R_x + R_N)^{-1} \left\{ (R_x + R_N) - R_x \right\}\right]$$

$$= \text{Trace } (R_x(R_x + R_N)^{-1} R_N) \qquad \text{.. (3.17)}$$

SUBOPTIMAL WIENER FILTERING (1) :

The concepts of generalised Wiener filtering can be utilised to perform suboptimal Wiener filtering in different transform domains. Here the problem is formulated as that of an unconstrained minimization of the m.s.e. with certain selected elements of the filter constrained to be zero. The main objective is to achieve computational convenience with filter performance as close to the optimum filter level as possible. One selects the zero elements in such a manner that fast computation is possible.

One example of this optimization is to use selected elements of the optimum Wiener filter matrix, given by equation

(3.14). For instance, only those terms which are of large magnitude can be retained. Another alternative will be to use only the diagonal and near diagonal terms of the optimum filter matrix. As can be seen from equation (3.14), the optimum Wiener filter matrix A is given by

$$\begin{aligned} A &= T\ (R_xH^t(HR_xH^t+(HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1})T^{-1} \\ &= (T.(R_xH^t)T^{-1})(T(HR_xH^t+(HR_xH^t) \odot R_{N_1} + R_{N_2})^{-1}\ T^{-1}) \\ &= (T(R_xH^t)T^{-1})(T(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})\ T^{-1})^{-1} \end{aligned}$$

One can use a diagonal filter A', as a special case of this suboptimal filtering, where

$$A'(i,i) = \frac{\left[T(R_xH^t)T^{-1}\right](i,i)}{\left[T(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})T^{-1})\right](i,i)} \qquad .. (3.18)$$

In fact, when T is a real transform and the filter is constrained to be a diagonal one, the minimization of m.s.e. leads to the filter A' given by equation (3.18). This can be proved as follows.

The restored output $\hat{x}$, as before (equation (3.3)) is given by

$$\hat{x} = T^{-1} A' \, T \, f$$

$$= T^{t} A' \, T \, f \qquad \text{.. (3.19)}$$

since T is a real unitary transform i.e. $TT^{t} = I$. In equation (3.19), the filter A' is assumed to be a diagonal one i.e.

$$A' = \begin{bmatrix} A'(1,1) & 0 & \cdots\cdots & 0 \\ 0 & A'(2,2) & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots\cdots & A'(M,M) \end{bmatrix}$$

From equation (3.2),

$$e^{2} = E\left[(x-\hat{x})^{t}(x-\hat{x})\right]$$

$$= E\,(x^{t}x) - 2\,E(x^{t}\hat{x}) + E(\hat{x}^{t}\hat{x})$$

$$= E\,(x^{t}x) - 2.E\,(x^{t}T^{t}\,A'T\,f) + E\,(f^{t}T^{t}\,A'^{t}\,T\,T^{t}.\;A'T\,f)$$

$$= E\left[x^{t}x\right] - 2E\left[(Tx)^{t}A'.(Tf)\right] + E\left[(Tf)^{t}.\;A'^{t}A'.\;(Tf)\right] \qquad \text{.. (3.20)}$$

The filter A' should be such that e^{2} is minimized.

$\therefore\ \nabla_{A'}(e^{2}) = 0$, where $\nabla_{A'}$, in this particular case is given by

$$\nabla_{A'}(e^2) = \begin{bmatrix} \frac{\partial(e^2)}{\partial A'(1,1)} & 0 & 0 & \cdots & 0 \\ 0 & \ddots & & & 0 \\ 0 & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & \cdots & & \frac{\partial(e^2)}{\partial A'(M,M)} \end{bmatrix}$$

Let $x_1 = Tx$, .. (3.21)

$f_1 = Tf$ .. (3.22)

Then, from (3.20), (3.21) and (3.22),

$$e^2 = E\ (x^t x) \quad - 2E\ (x_1{}^t A' f_1)\ + E(f_1{}^t\ A'^t\ A' f_1)$$

Now, $\nabla_{A'} \left\{ E\ (x_1{}^t A' f_1) \right\} \quad = \nabla_{A'} \left\{ E(\sum_{i=1}^{M} {x_1}_i{}^{A'(i,i)} {f_1}_i) \right\}$

(where, for a vector R, R_i denotes the ith element)

$$= D \left[E(x_1 f_1{}^t) \right]$$

[where D is an operator, which,when operated on a matrix takes the diagonal elements and produces the corresponding diagonal matrix.]

$$= D.\left[E(Txf^t\ T^t)\right] \ = D\left[T.E(xf^t).\ T^t\right]$$

$$= D\left[T.R_x H^t T^t\right] \ \ldots (3.23) \quad (\text{ From equation } 3.12)$$

Similarly,
$$\nabla_{A'} \left\{E\ (\ F_1{}^t\ A'^t f_1)\right\}$$

$$= \nabla_{A'} \left\{E\ (\sum_{i=1}^{M} A'(i,i)^2\ \ f_{1_i}{}^2)\right\}$$

$$= \ 2.D\left[\ E\ (A'\ f_1 f_1{}^t\ )\ \right]$$

$$= \ 2.D\left[\ E(A'\ T\ ff^t\ T^t)\right]$$

$$= \ 2.D\left[A'\ \ T.\ \ E\ (ff^t)\ T^t\right]$$

$$= 2.D\left[A'\ T\ \ (HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})\ T^t\right]$$

(From equation (3.13))

$$= 2A'.D\left[T(HR_xH^t + (HR_xH^t) \odot R_{N_1} + R_{N_2})\ T^t\right]$$

.. (3.24)

(Since A' itself is a diagonal matrix)

Hence from equations (3.20), (3.23) and (3.24) we get

$$\nabla_{A'} (e^2) = 0$$

or, $$2.A'.D\left[T\,(HR_xH^t + (HR_xH^t)\odot R_{N_1} + R_{N_2})\,T^t\right]$$

$$= 2.D\left[T.\,R_xH^tT^t\right]$$

which leads to the following equation for the diagonal elements of the diagonal matrix A' :

$$A'(i,i) = \frac{\left[T.(R_xH^t)T^{-1}\right](i,i)}{\left[T(HR_xH^t + (HR_xH^t)\odot R_{N_1} + R_{N_2})T^{-1}\right](i,i)}$$

$$\left[\text{Since } T^{-1} = T^t\right]$$

which is the suboptimal filter already described by equation (3.18). The resulting filter is computationally convenient because (i) it is a diagonal filter and so filter matrix multiplication is less complicated. (ii) it avoids matrix inversion.

Iterative generalised Wiener filtering : - The discrete, generalised Wiener filter, like the continuous one can be criticised on the following grounds :

(i) The filter, as given by equation (3.14) requires an extensive knowledge of the autocorrelation matrix of the original object. Though the autocorrelation properties of the noise can be obtained from a suitable model, that of the object is essentially an intrinsic property of the object itself.

(ii) The Wiener filter produces the optimum estimate only on an average and there is no guarantee that every filter operation will produce the optimum restoration.

(iii) Since the Wiener filter solves an unconstrained minimization problem, false details due to negative pixel values are also encountered.

The same iterative technique, which was proposed to avoid the above mentioned problems in the case of continuous Wiener filter, works out here also.

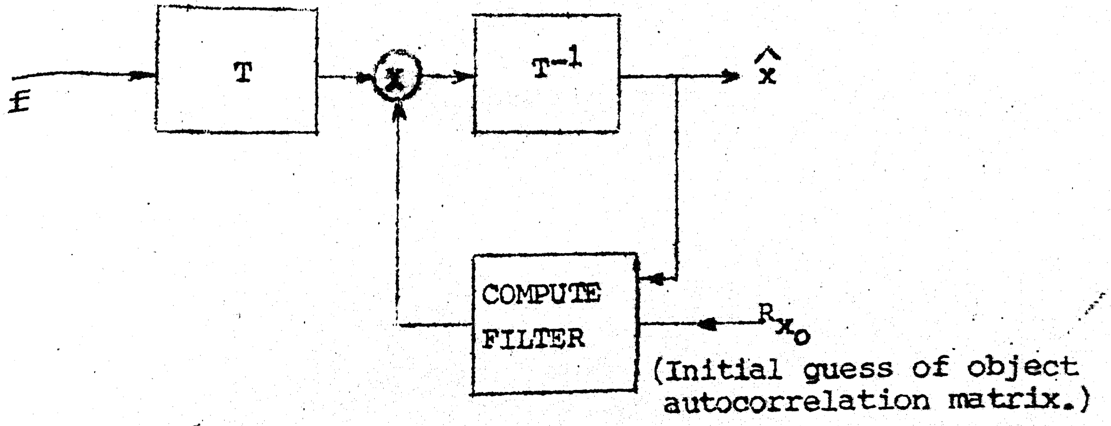


Fig. 3.3 Iterative generalised Wiener filtering

Thus one starts with an initial guess R_{x_o} of R_x, obtains an estimate, uses it to compute a better (hopefully) object autocorrelation matrix and proceeds in an iterative manner till significant improvement results. Also note that the iterative procedure can be applied in the case of suboptimal filtering also, since, from equation (3.18), it is seen that the diagonal filter A' also requires the precise knowledge of R_x.

The iterative Wiener filter does not require the extensive information about object autocorrelation matrix. In addition, non-negativity of the pixel values may be introduced in the steps of iteration. Moreover, the whole algorithm is tied to the given recorded image and so reasonable restoratiom may be expected for each object (and not just on the average).

CHAPTER 4

SIMULATION & RESULTS

SIMULATION OF THE DEGRADED AND NOISY IMAGE :

The proposed scheme for iterative, generalised discrete Wiener filtering was implemented with two 16 x 16 images, blurred by two row and column convolution matrices H_R and H_C, where

$$H_C^t = H_R = \begin{bmatrix} 1 & 0 & 0 & \cdots & & & \frac{1}{4} \\ \frac{1}{4} & 1 & 0 & \cdots & & & 0 \\ 0 & \frac{1}{4} & 1 & \cdots & & & 0 \\ \vdots & \vdots & & & & & \vdots \\ 0 & & & & \frac{1}{4} & 1 & 0 \\ 0 & \cdots & & & 0 & \frac{1}{4} & 1 \end{bmatrix}$$

If P is the original picture matrix, then the blurred matrix P' is given by

$$P' = H_R.P.H_C \qquad .. (4.1)$$

If v be the vector obtained from the picture matrix P by stacking the elements columnwise i.e.,

$$v = \Big[P(1,1),\ P(1,2),\ \ldots.\ P(1,M),\ P(2,1),\ P(2,2),\ \ldots.\ P(2,M),\ \ldots.\ P(M,M-1),\ P(M,M)\Big]^t \qquad ..\ (4.2)$$

then the corresponding degraded output vector v' (which can also be obtained from P' by similar process of columnwise stacking) is given by (3)

$$v' = H.v \qquad ..\ (4.3)$$

where,

$$H = H_R \otimes H_C^t \qquad ..\ (4.4)$$

$\otimes$ denotes left direct (Kronecker) product of two matrices i.e. if A and B be two MXM matrices, then

$$A \otimes B = \begin{bmatrix} B_{11}[A] & B_{12}[A] & \ldots. & B_{1M}[A] \\ B_{21}[A] & B_{22}[A] & \ldots. & B_{2M}[A] \\ \vdots & \vdots & & \vdots \\ B_{M1}[A] & B_{M2}[A] & \ldots. & B_{MM}[A] \end{bmatrix}$$

If A and B are circulant matrices, then A $\otimes$ B is a block circulant matrix. Note that, in general in an imaging system, one may not be able to separate out the row and columnwise degradation processes. In such a case H (equation (4.4)) can not be expressed as a direct product of row and columnwise degradation matrices and the blurring is described by equation (4.3). Thus equation (4.3) is the most general representation of image degradation.

Zero-mean, unit-variance, white, Gaussian noise is added to the degraded picture. Moreover, the picture is also corrupted by zero-mean, white, Gaussian multiplicative noise. The noise auto-correlation matrix is clearly a diagonal matrix, since for a zero-mean, white noise process,

$$E(n_i \, n_j) = \sigma^2 . \delta_{ij} \qquad .. \; (4.5)$$

where n_i and n_j are the i-th and j-th samples of the white noise and σ^2 is the noise variance.

OBJECT AUTOCORRELATION AND ITS COMPUTATION :

The autocorrelation R(i,k) of a discrete random process x(i) is given by

$$R(i,k) = E(x(i).x(i+k)) \qquad .. \; (4.6)$$

If the discrete sequence vector x, of which x(i) is the i-th component belongs to a second order stationary process [We define $x \triangleq (x(1), x(2), \ldots . \; x(M))^t$], then autocorrelation becomes a function of k. For such a stationary process, equation (4.6) modifies to,

$$R(k) = E(x(i).x(i+k)) \qquad .. \ (4.7)$$

for any i.

The object auto-correlation matrix R_x is given by

$$R_x = E(xx^t)$$

$$= \begin{bmatrix} E(x(1).x(1)) & E(x(1).x(2)) & .\ .\ .\ . & E(x(1)x(M)) \\ E(x(2).x(1)) & E(x(2).x(2)) & .\ .\ .\ . & E(x(2).x(M)) \\ \vdots & \vdots & & \vdots \\ E(x(M).x(1)) & E(x(M).x(2)). & .\ .\ . & E(x(M).x(M)) \end{bmatrix}$$

Since $E(x(i).x(j)) = E(x(j).x(i))$, R_x is a symmetric matrix. From this and the assumption of stationarity of x, it is evident that R_x takes the following structure.

$$R_x = \begin{bmatrix} R(0) & R(1) & R(2) & .\ .\ .\ .\ . & R(M-1) \\ R(1) & R(0) & R(1) & R(2) & . \\ R(2) & R(1) & \searrow & \searrow & . \\ \vdots & R(2) & \searrow & \searrow & \vdots \\ R(M-1) & & \searrow & \searrow & \\ & .\ .\ .\ . & & & \end{bmatrix} \qquad .. \ (4.8)$$

(By the arrows, we mean that the elements R(0), R(1), R(2) .... R(M-1) are continued in the corresponding lines.)

Such a matrix is a symmetric Toeplitz matrix. For a two-dimensional object, both rowwise and columnwise autocorrelations take part in the object autocorrelation. With v being the vector obtained from the picture matrix P by stacking the elements columnwise (equation (4.2)), the corresponding autocorrelation matrix $R_{vv} = E(vv^t)$ takes the following form.

$$R_{vv} = \begin{bmatrix} [B_0] & [B_1] & \cdots & \cdots & [B_{M-1}] \\ [B_1] & [B_0] & [B_1] & & \vdots \\ \vdots & [B_1] & \searrow & \searrow & \vdots \\ \vdots & & \searrow & \searrow & \vdots \\ [B_{M-1}] & \cdots & \cdots & & \end{bmatrix}$$

where B_i denotes the i-th MXM block. Such a matrix is known as a block-Toeplitz matrix. Here each block is a MXM Toeplitz matrix because of rowwise stationarity and the blocks are arranged in Toeplitz form because of columnwise stationarity. Note that each block B_i , i = 0,1, ... (M-1), incorporates the autocorrelation between two rows at a gap of i.

Also note that if the object autocorrelation can be separated both row and columnwise , then R_{vv} can be expressed as

$$R_{vv} = R_R \otimes R_C \qquad .. (4.10)$$

where, as before $\otimes$ denotes the left direct product of matrices and R_C and R_R are the columnwise and rowwise autocorrelation matrices respectively.

The computation of the obj ect autocorrelation matrix is based on the assumption of ergodicity of the data sequence. The sample autocorrelation function R(k) is computed by the following equation :

$$R(k) = \frac{1}{n} \sum_{i=0}^{n-1} x_i \, x_{i+k} \qquad .. (4.11)$$

where $\quad n = M-k \qquad .. (4.12)$

(M is the length of the sequence.)

Now, consider a two-dimensional sequence x, given by

$$x = \begin{bmatrix} x_{0,0} & x_{0,1} & \cdots\cdots & x_{0,M-1} \\ x_{1,0} & x_{1,1} & \cdots\cdots & x_{1,M-1} \\ \vdots & & & \\ \vdots & & & \\ x_{M-1,0} & x_{M-1,1} & \cdots\cdots & x_{M-1,M-1} \end{bmatrix}$$

The corresponding autocorrelation $R(k_1,k_2)$ i.e. autocorrelation between samples at a separation (k_1,k_2) is computed from the following equation :

$$R(k_1, k_2) = \frac{1}{mn} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} x_{i,j}\, x_{i+k_1,\, j+k_2} \qquad .. \ (4.13)$$

where

$$m = M-k_2 \qquad .. \ (4.14)$$

$$n = M-k_1 \qquad .. \ (4.15)$$

It is to be noted that for stationary data, the computation of the autocorrelation matrix requires the evaluation of any row, or, column only, because of the symmetric Toeplitz (or block-Toeplitz in case of two-dimensional signals) structure of the autocorrelation matrix. Thus one can evaluate the autocorrelation values for different values of k (or k_1, k_2 for 2D data) using equation (4.11) (or (4.13) for 2D data) and thereby obtain the autocorrelation matrix.

The proposed scheme of iterative, generalised, Wiener filtering was applied to two 16 x 16 images. The initial guess for object autocorrelation was derived from the recorded data itself. With input SNR reasonably high and degradation reasonably less, the choice is quite appropriate. Also one can start with a suitable model for the object autocorrelation and compute the several associated parameters from the given data (e.g. in many situations, it is quite appropriate to use a low pass filter function as the object PSD, or equivalently, an exponential autocorrelation function). The mean of the object was computed from the iterates in each step of iteration and substracted from the iterates themselves, thereby generating a zero-mean input.

RESULTS

The simulation results corresponding to the two images are described below.

(A) The corresponding image is shown in Fig. 4.1(a). The noisy and degraded image is shown in Fig. 4.1(b).

Variance of additive, white, Gaussian noise = 1.0

Variance of multiplicative, white, Gaussian noise = 0.1

Initial mean square error = 18.47

(i) Optimal space domain filter

$$A = R_x H^t (HR_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2})^{-1}$$

No. of Iterations	1	2	3	4	5	6
Mean square error	22.15	15.61	12.00	10.13	8.01	7.91

(ii) Suboptimal filtering

$$A(i,i) = \frac{[T R_x H^t T^{-1}]\ (i,i)}{[T(HR_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2})\ T^{-1}]\ (i,i)}$$

where $T^{-1} = T^{*t}$ and A is a diagonal matrix.

(a) $T = I$

No. of iterations	1	2	3	4
Mean square error	11.01	10.00	9.89	9.76

(b) T = Hadamard transform

No. of iterations	1	2	3	4	5
Mean square error	11.79	10.62	9.82	8.93	8.89

(c) T = FFT

No. of iterations	1	2	3	4	5	6
Mean square error	13.26	11.85	11.03	10.69	10.01	9.46

The pictures corresponding to the different steps of iteration are shown in Figs. 4.2 - 4.5.

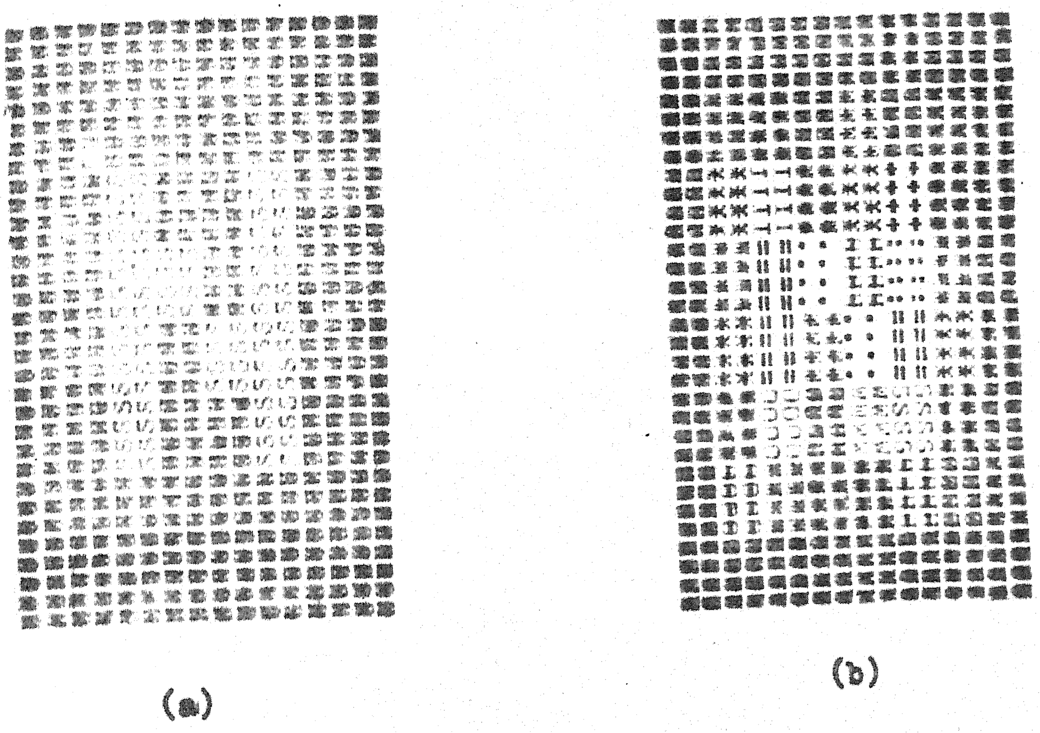
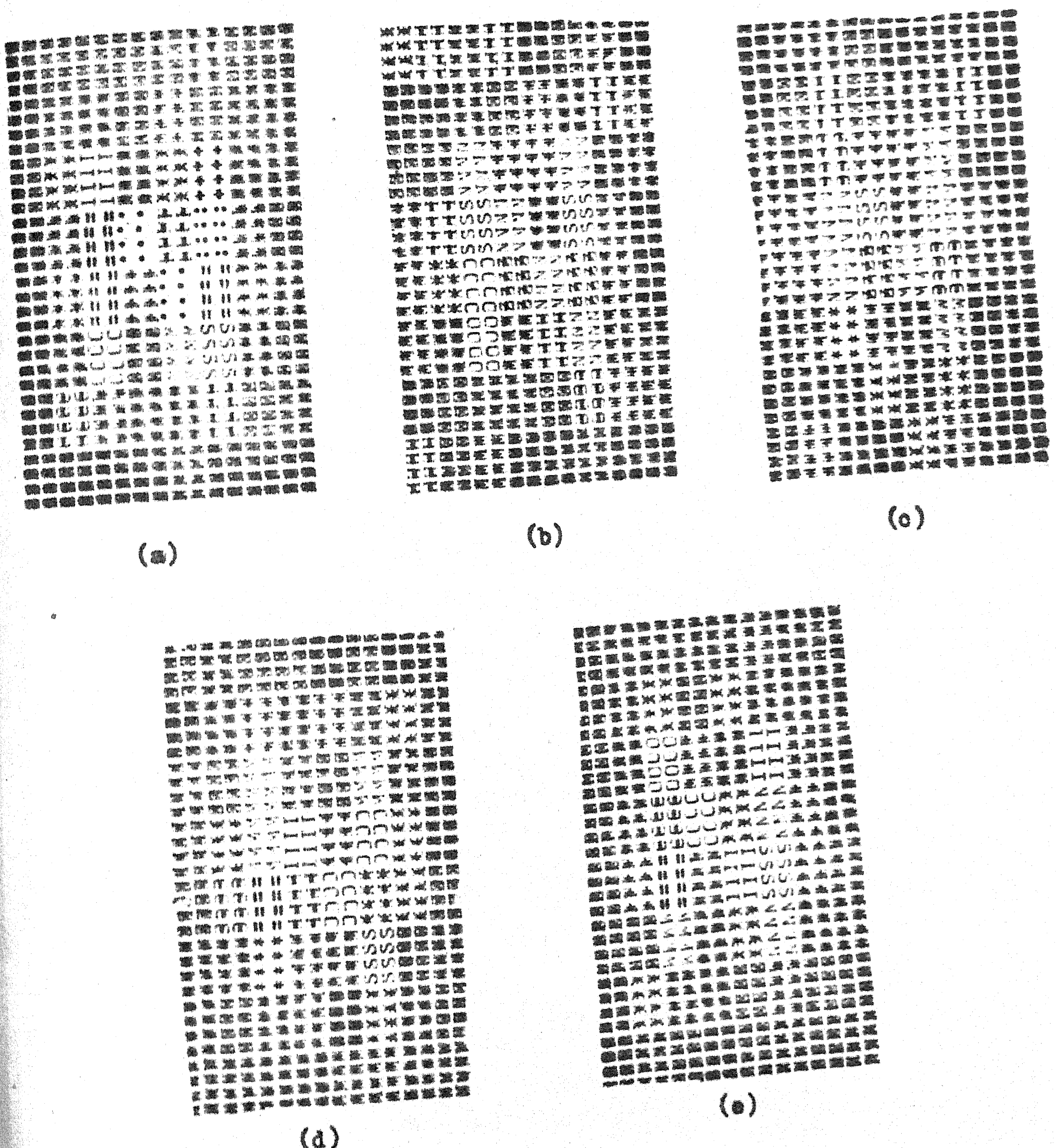


Fig. 4.1 (a) Original image ,
(b) Noisy and degraded image.



(a)

(b)

(c)

(d)

(e)

Fig. 4.2 First five iterates ((a) to (e)) when

$$A = R_X H^t (HR_X H^t + (HR_X H^t) \circ R_{N_1} + R_{N_2})^{-1}$$

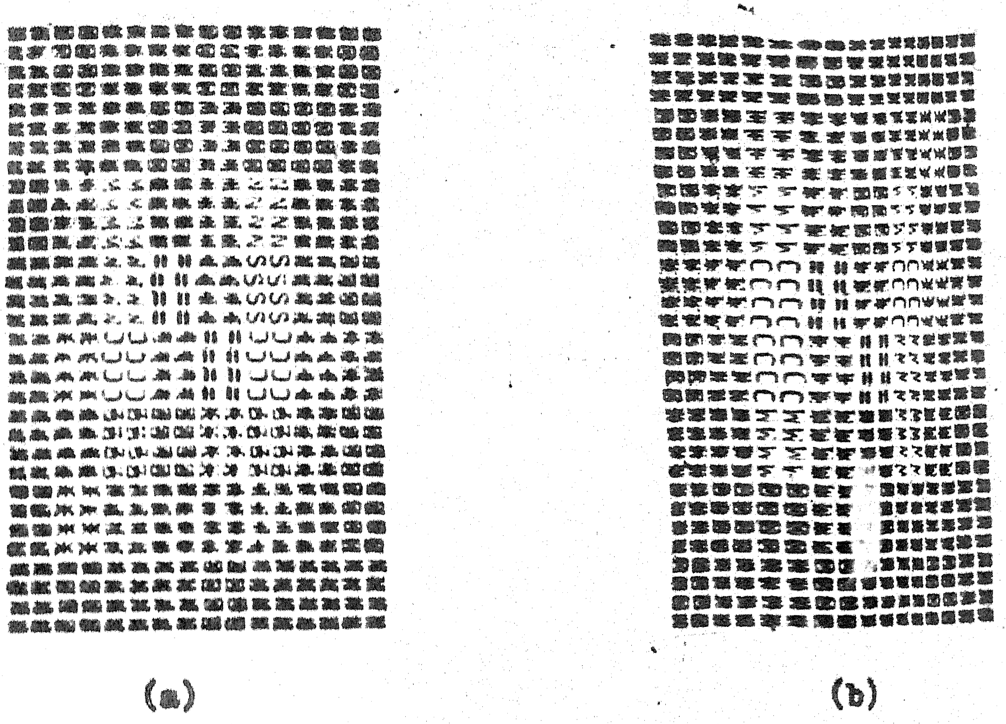


Fig. 4.3 First two iterates ((a)-(b)) for suboptimal filtering in space domain.

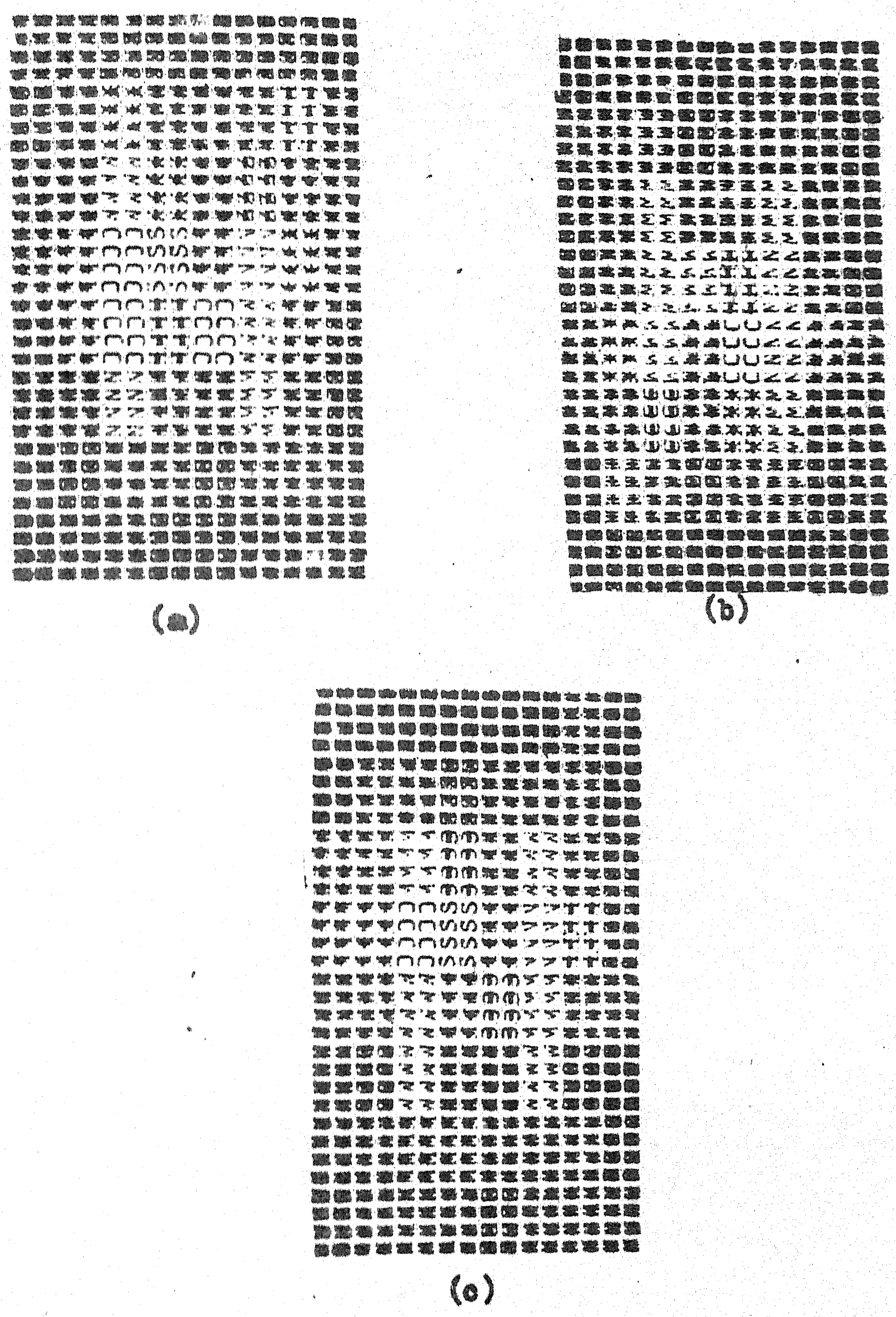


Fig. 4.4 First three iterates ((a) - (c)) for suboptimal filtering in Hadamard domain.

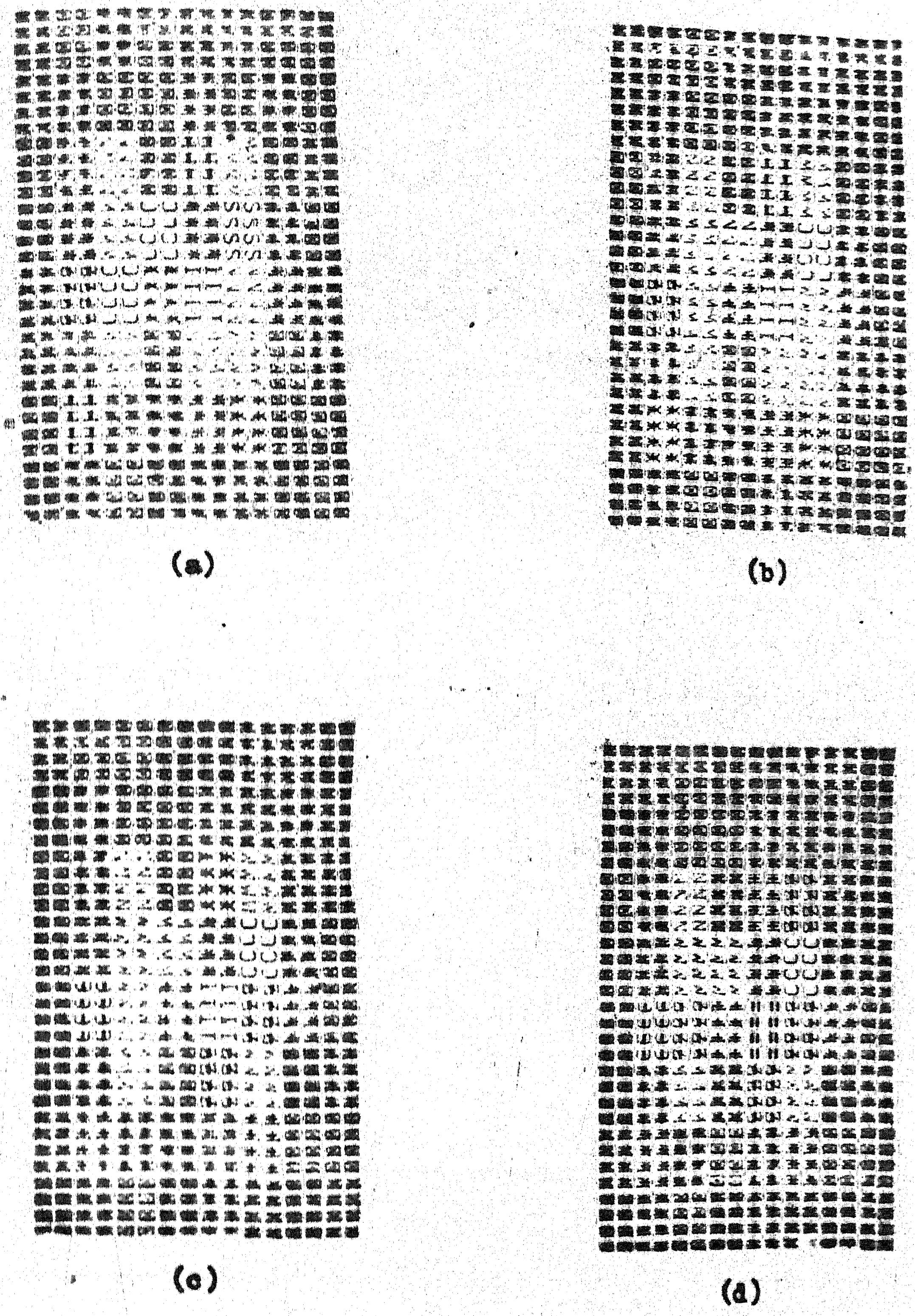


Fig.-4.5 First four iterates ((a) - (d)) for suboptimal filtering in FFT domain.

(B) The original image and the corresponding noisy and degraded picture are shown in Fig. 4.6(a) and Fig. 4.6(b) respectively.

Variance of additive, white, Gaussian noise = 1.00

Variance of multiplicative, white, Gaussian noise = 0.15

Initial mean square error = 31.34

(i) Optimal space domain filter

$$A = R_x H^t (HR_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2})^{-1}$$

No. of iterations	1	2	3	4	5	6
Mean square error	25.16	12.50	11.99	10.26	9.87	8.71

(ii) Suboptimal filtering

$$A(i,i) = \frac{[T R_x T^{-1}]\,(i,i)}{[T(HR_x H^t + (HR_x H^t) \odot R_{N_1} + R_{N_2})^{-1} T^{-1}]\,(i,i)}$$

where $T^{-1} = T^{*t}$ and A is a diagonal matrix.

(a) $T = I$

No. of iterations	1	2	3	4	5
Mean square error	16.31	15.71	15.53	15.17	15.15

(b) T = Hadamard transform

No. of iterations	1	2	3	4	5
Mean square error	14.86	12.98	10.75	9.78	9.81

(c) T = FFT

No. of iterations	1	2	3	4	5	6
Mean square error	16.45	14.86	13.52	13.57	12.33	10.37

The corresponding pictures are shown in Figs. 4.7-4.10.

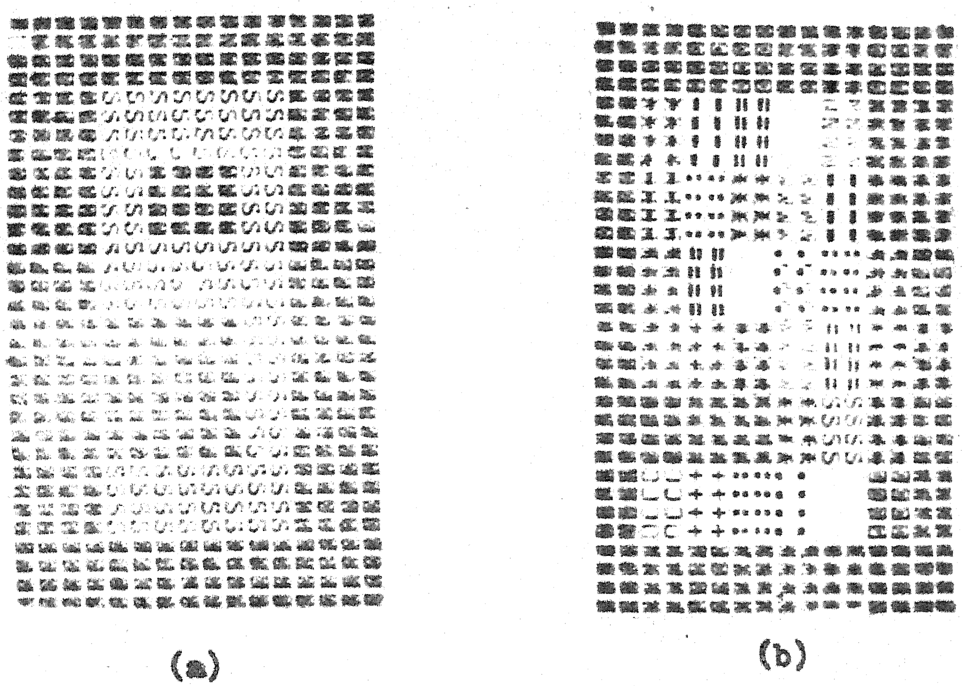
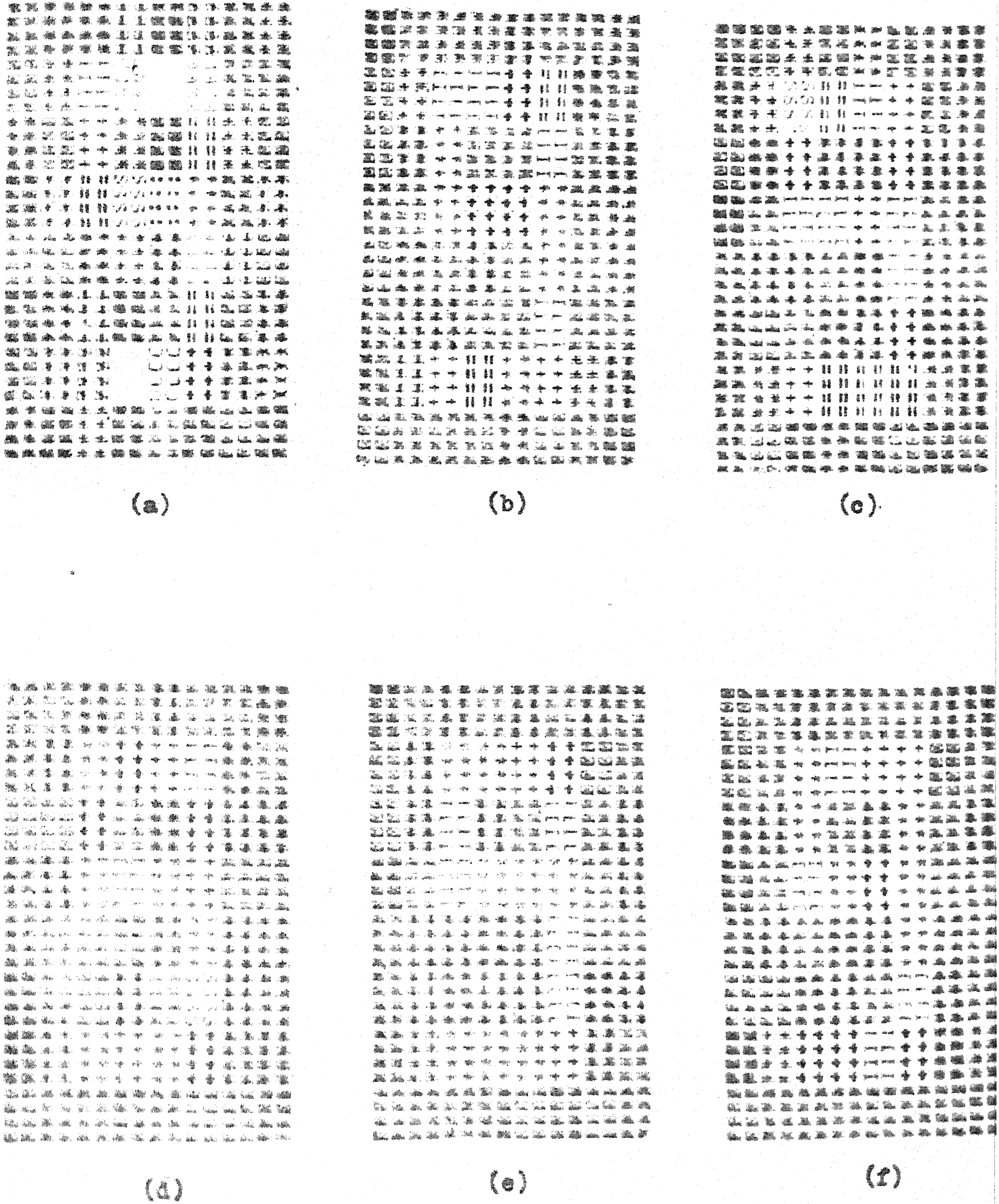


Fig. 4.6 (a) Original image,
(b) Noisy and degraded image.



(a) (b) (c)

(d) (e) (f)

Fig. 4.7 First six iterates ((a) - (f))
when $A = R_X H^t (HR_X H^t + (HR_X H^t) \odot R_{N_1} + R_{N_2})^{-1}$.

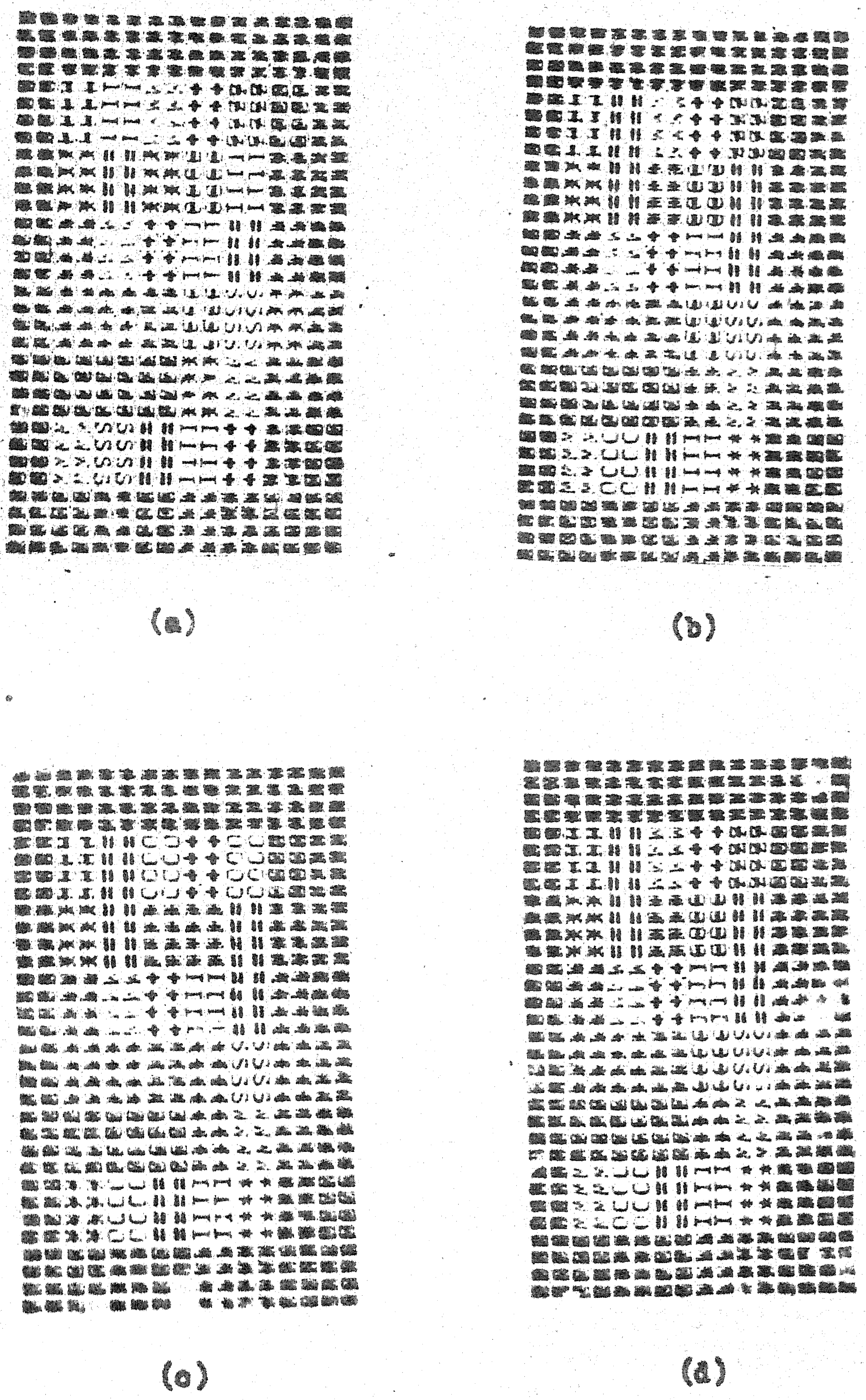
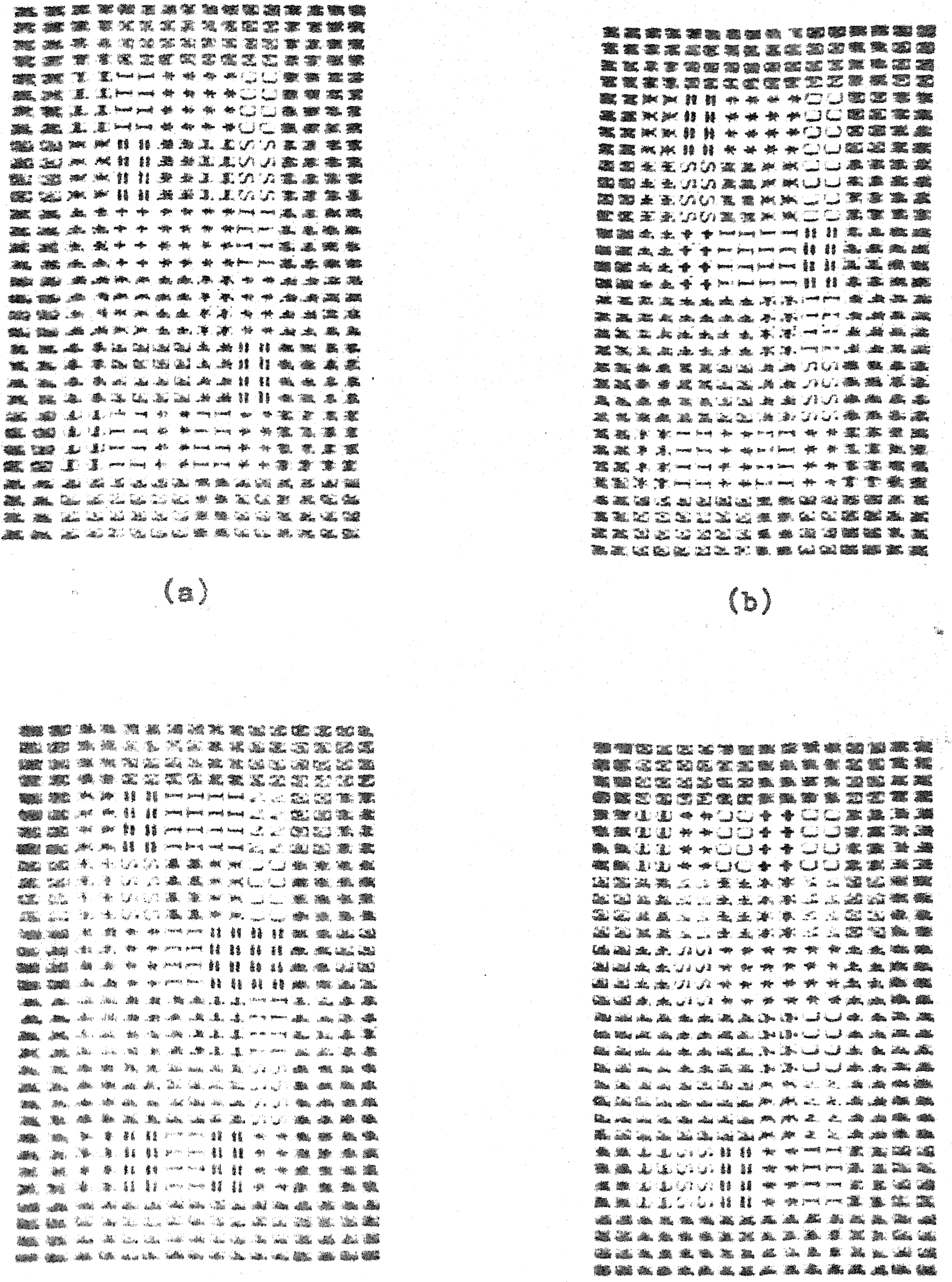


Fig. 4.8 First four iterates ((a).- (d)) for suboptimal filtering in space domain.



(a)

(b)

(c)

(d)

Fig. 4.9 First four iterates ((a) - (d)) for suboptimal filterin in Hadamard domain.

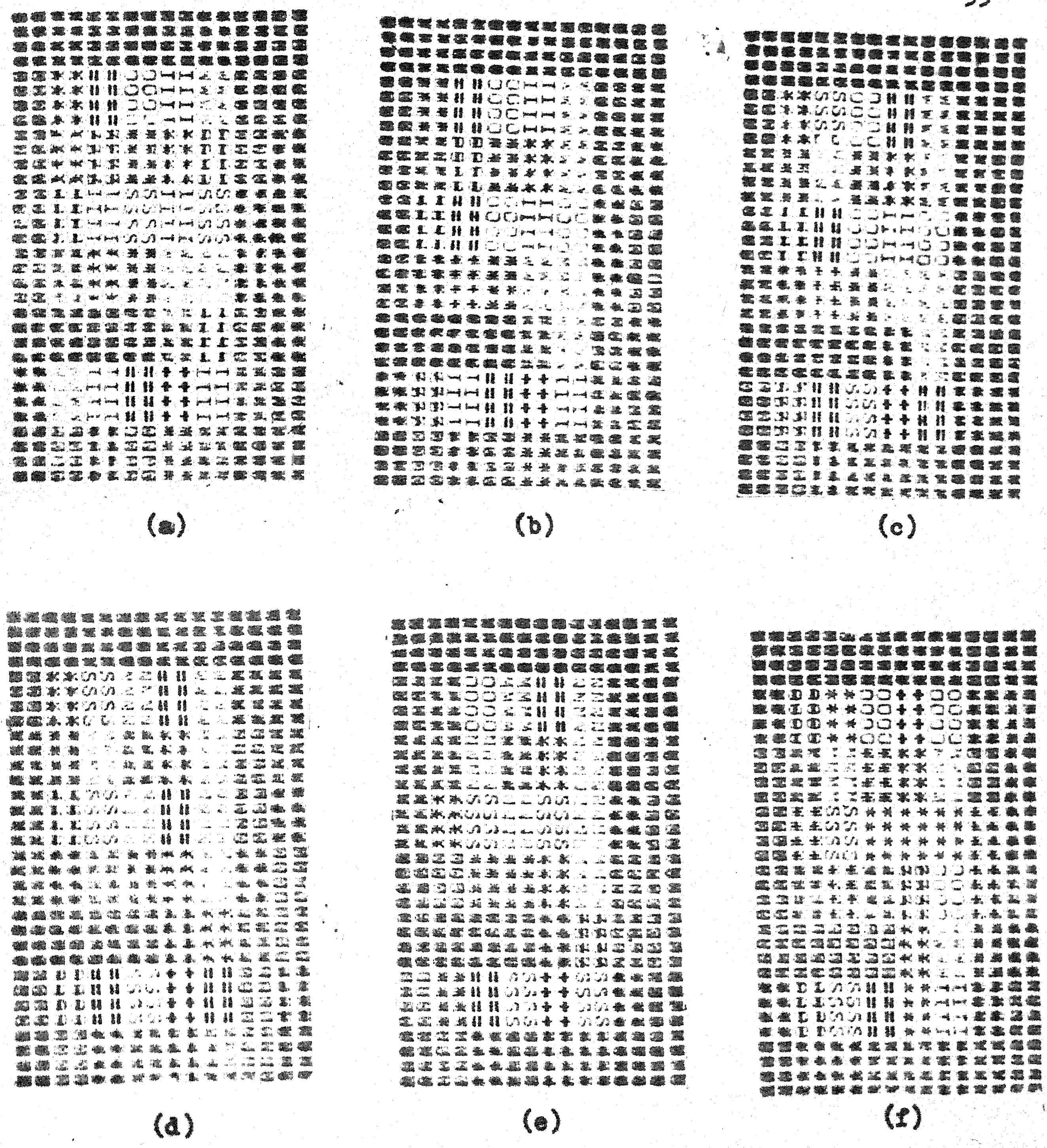


Fig. 4.10 First six iterates ((a) - (f)) for suboptimal filtering in FFT domain.

CHAPTER 5

CONCLUSION

An attempt has been made in this thesis to remedy some of the shortcomings of conventional Wiener filter through an iterative algorithm. The limited simulation studies of the algorithm for the case of generalised Wiener filtering have shown the effectiveness of the iteration strategy. Mathematical proof of the convergence of the algorithm is a formidable task and it is felt that more work is required to solve the problem.

The restoration problem can be simplified with the aid of circulants for space-invariant degrading systems. The concept of asymptotic equivalence of circulants and Toeplitz matrices can be used to approximate Toeplitz (Block-Toeplitz) correlation matrices by circulants (Block-circulants). The latter permit the use of FFT-algorithms and the problem becomes computationally convenient.

It is observed that for suboptimal filtering, the iterative algorithm works better in the case of processing in Hadamard domain than in FFT domain. This can be attributed to the fact that the pictures used for simulation are binary in nature i.e. they have only two levels, high and low and so their representation and processing in Hadamard domain leads to less loss of information in comparison to FFT. It is expected that for pictures with continuous variation of gray levels, FFT-domain processing would work better.

REFERENCES

1. W.K. Pratt, 'Generalised Wiener Filtering Computation Techniques', IEEE Trans. on Computers C-21(7) 636-641, (1972).

2. Ramakrishna, R.S., 'Some Iterative Techniques in Digital Image Restoration' Ph.D. thesis, Dept. of Elect. Engg., Indian Institute of Technology, Kanpur, 1978.

3. W.K. Pratt, 'Digital Image Processing', John Wiley and Sons, Inc., 1978.

4. Rosenfeld, A., Kak, A.C., Digital Picture Processing, Academic Press, Inc., New York, 1976.

5. Papoulis, A., 'Signal Analysis' McGraw Hill Book Co., New York, 1977.

6. Devenport, W.B., Root, W.L., 'An Introduction to the Theory of Random Signals and Noise', McGraw Hill Book Co., New York, 1972.

7. Ramakrishna, R.S., Mullick, S.K., Rathore R.K.S., 'Iterative Image Restoration' IEEE Proc. of the International Conf. on System, Man and Cybernetics, 1983 (Delhi); Vol. II, pp. 1088-1091.

8. Helstrom, C.W., 'Image Restoration by the Method of Least Squares' Journal of the Optical Society of America, Vol. 57, pp. 297-303 ; 1967.

9. Van Trees, H.L. 'Detection, Estimation and Modulation Theory' Vol. I, John Wiley and Sons., Inc. New York, 1968.

APPENDIX

In chapter 3, we mentioned that

(i) $E(H\,xx^t\,H^t\,N_1) = [0]$

(ii) $E(N_1H\,xx^t\,H^t) = [0]$

(iii) $E(N_1\,Hx\,n_2^t) = [0]$

(iv) $E(n_2x^t\,H^t\,N_1) = [0]$

where, as described in chapter 3, the original object x and the additive and multiplicative noise components i.e. n_2 and N_1 are mutually independent, zero-mean random processes. Here we give a formal proof of these results.

(i) $E\,(Hxx^t\,H^t\,N_1)_{ij}$

$$= E\,(\sum_l \sum_k H_{il}\,x_l\,x_k\,N_{1_{jj}}\,H_{jk})$$

$$= (\sum_l \sum_k H_{il} . E\,(x_l\,x_k) . E(N_{1_{jj}})\,H_{j_k})$$

$= 0$, since the object x and both N_1 and N_2

are zero-mean, mutually independent random processes.

Hence $E(Hxx^tH^tN_1) = [0]$

(ii) $E(N_1 \; H \; xx^t \; H^t)_{ij}$

$$= E(\sum_l \sum_k N_{1_{ii}} \; H_{il} \; x_l \; x_k \; H_{jk})$$

$$= (\sum_l \sum_k E(N_{1_{ii}}) . H_{il} . E(x_l x_k) \; H_{jk})$$

$= 0$; Hence $E(N_1 H \; xx^t \; H^t) = [0]$

(iii) $E(N_1 \; Hx \; n_2{}^t)_{ij}$

$$= E(\sum_k N_{1_{ii}} \; H_{ik} . \; x_k \; n_{2_j})$$

$$= (\sum_k E(N_{1_{ii}}) . \; H_{ik} . \; E(x_k) . \; E(n_{2_j}))$$

$= 0$

Hence $E \; (N_1 H x n_2{}^t) = [0]$

(iv) $E(n_2 x^t H^t N_1)_{ij}$

$$= E(\sum_k n_{2_i} \; x_k \; N_{1_{jj}} \; H_{jk})$$

$$= (\sum_k E(n_{2_i}) . \; E(x_k) . \; E(N_{1_{jj}}) . \; H_{jk})$$

$= 0$

Hence $E(n_2 x^t H^t N_1) = [0]$